



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Laboratorio di Sicurezza Informatica

Autenticazione

Marco Prandini

Dipartimento di Informatica – Scienza e Ingegneria

La regola AAA

- **Autenticazione:** attribuzione certa dell'identità di un soggetto che utilizza le risorse
 - Normalmente include una *identificazione* preliminare
 - È una separazione importante!
 - Errore comune usare elementi *identificativi* come “segreti” che supportano una *autenticazione* solo perché sembrano “oscuri”
- **Autorizzazione:** verifica dei diritti di un soggetto di compiere una determinata azione su di un oggetto
 - decisione esplicita di concessione o negazione del permesso
- **Auditing:** tracciamento affidabile delle decisioni (tutte) di autenticazione e autorizzazione
 - verifica dell'efficacia delle politiche
 - compromesso difficile sui dettagli tra utilità e usabilità

Autenticazione

- L'autenticazione è basata sull'utilizzo di uno di questi fattori - qualcosa che solo l'utente:
 - **Conosce** (ad es. Password, PIN, risposta segreta)
 - **Possiede** (ad es. Carta bancomat, telefono cellulare, hard token, Yubikey)
 - **È (fisicamente)** (ad es. Biometrico: iride, impronta digitale, ecc.)
 - **È (posizione)** (ad es. GPS, geolcation, Centralizzata Auto, Allarme in casa.)
- Soffermiamoci sui modi che un **Prover** ha per dimostrare a un **Verifier** di essere a conoscenza di un segreto

Autenticazione passiva

■ Autenticazione *passiva*

- P e V concordano il segreto e lo memorizzano
- P invia il segreto a V per dimostrare di conoscerlo
- V lo confronta con la copia in suo possesso per autenticare P

■ Problemi di comunicazione

- invio in chiaro → intercettazione da parte di attaccante passivo
- invio offuscato ma sempre uguale → replay attack

servono protocolli di autenticazione più sofisticati

- ... oppure un canale cifrato bene

■ Problemi di memorizzazione

- furto da V

Memorizzazione delle password

■ Requisiti

- V non deve conoscere le password
 - Il furto di file da V deve essere inefficace
 - V deve discriminare una password corretta da una errata
- non si memorizza la password ma la sua impronta, calcolata con una funzione hash

■ Problemi

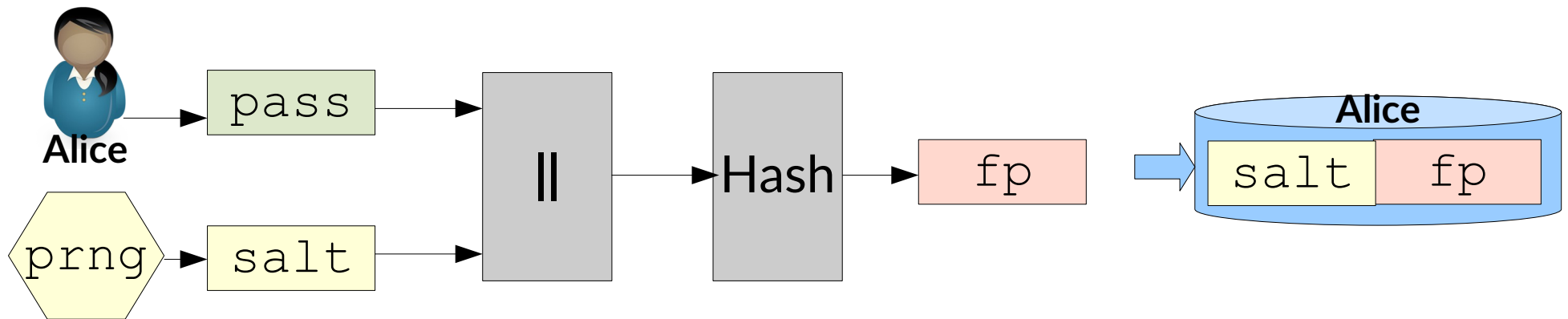
- attacco con dizionario
 - vedi approfondimento su Rainbow Tables
- stessa password su macchine diverse

■ Mitigazione: salt

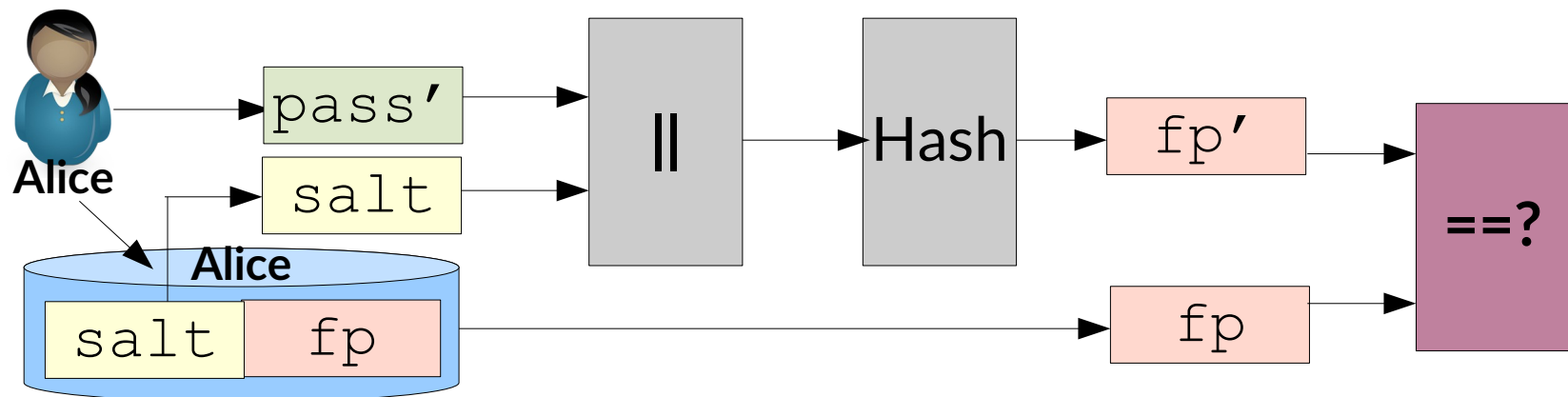
- variazione random inserita alla scelta della password

Salt

■ Scelta della password



■ Verifica della password



Salt

■ Esempio da /etc/shadow

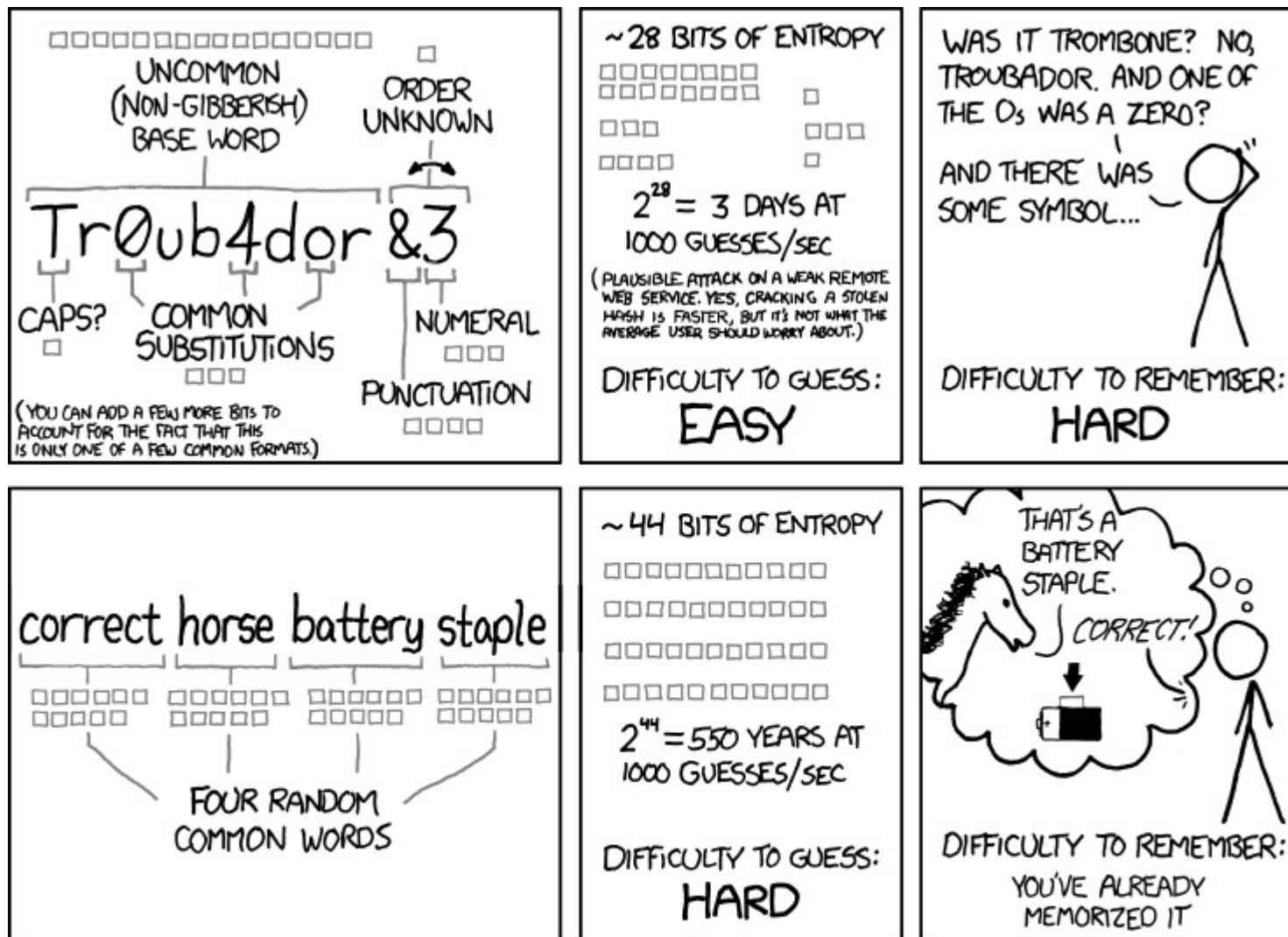
```
$6$ViDM2ltuaSNPBxfO$lp40UoauO.OiFafalVeMazZplisBV.CC76j  
Ryead9rvidbyWcAr200dH.7N8budnpS3FzJJdDxrKGQjekwTFU0
```

identificatore dell'algoritmo hash usato
salt

fingerprint calcolato su concatenazione pass||salt

- a ogni scelta o rinnovo della password, il salt cambia
 - stessa pass, fingerprint diverse
 - non permette di precalcolare le fingerprint partendo da un dizionario
- è inefficace contro attacchi offline, cioè avviati dopo aver sottratto il file delle password
 - la contromisura essenziale per questi è che la password non sia facile da indovinare

Scelta delle password



THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

Protezione delle password

- In ogni caso, una password scoperta grazie a un lungo lavoro di cracking su di un *leak* da un sistema, sarà usabile ovunque
- Bisogna usare password diverse a prescindere
 - decine? centinaia??
 - *password manager*
 - database cifrato con passphrase
 - quella va scelta bene
- Rischio mitigato da sistemi a più fattori
 - vedi seguito

Cattive pratiche

Modifica Password

Password Attuale

Nuova Password

Conferma Nuova Password

La password deve contenere almeno un carattere numerico

Annulla Conferma

Conferma Nuova Password

La password è troppo lunga

Annulla

Conferma

Cattive pratiche

```
prandini@disi057118:~$ pass generate aa.com 15
An entry already exists for aa.com. Overwrite it? [y/N] y
[master d79a692] Add generated password for aa.com.
  1 file changed, 0 insertions(+), 0 deletions(-)
  rewrite aa.com.gpg (100%)
The generated password for aa.com is:
]09y>F{uh7*gHt:
prandini@disi057118:~$
```

First foreign

ecurity ques

City where y

ecurity ques

Make and m

Password requirements

- 6-16 characters
- Any combination of special characters, letters and numbers
- No spaces before the first, or after the last characters

(• Required)

Current password •

.....

❗ New password •

.....

Invalid password.

❗ Confirm password •

.....

Invalid password.

Cattive pratiche

Hertz GOLD PLUS
REWARDS

CHIUDI ✕

Reset Your Password

Password:*

Please retype your password to confirm:*

Reset Password

Cattive pratiche

1 Si è verificato un errore. Prima di proseguire tenga presente quanto segue:

✖ Your password is limited to the following special characters: !"#\$%&'()*+,-./:;<?_@ (60062).

Registrato dal

10/07/2017

Vecchia password


.....

Password

.....



Cattive pratiche (collaterali)



EUROPEAN CYBER SECURITY ORGANISATION

ecsportal.ecs-org.eu says

That user name already exists. Please choose another one.

OK

HOME
WEBSITE

MARCO PRANDINI
LOG OUT

My ECSO

- My calendar
- My Working Groups / My Bodies
- My downloads
- My notifications
- My details
- My user name and password**

My user name and password

My ECSO > My user name and password

User name:	<input type="text" value="marco.prandini@unibo.it"/>
Current password:	<input type="password" value=""/>
New password:	<input type="password" value=""/>
The password must contain at least 6 characters	
Confirm new password:	<input type="password" value=""/>

Discard changes

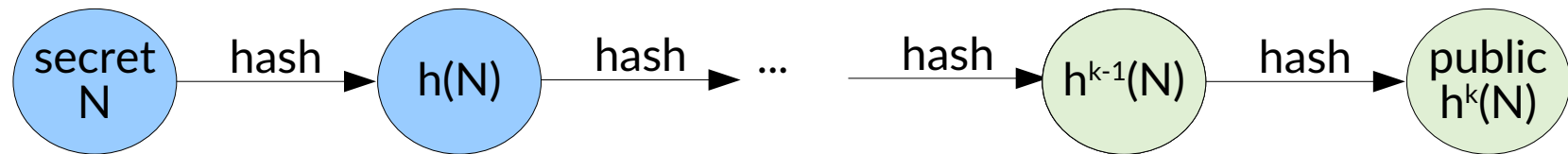
Save changes

Autenticazione attiva

- P convince V di possedere il segreto autentico senza svelarlo e mandando ogni volta un dato diverso
 - Il furto del dato di confronto da V è intrinsecamente inutile
 - Il furto dal canale è inutile per autenticazioni future
 - attenzione comunque all'uomo nel mezzo!

S-KEY One-Time Password

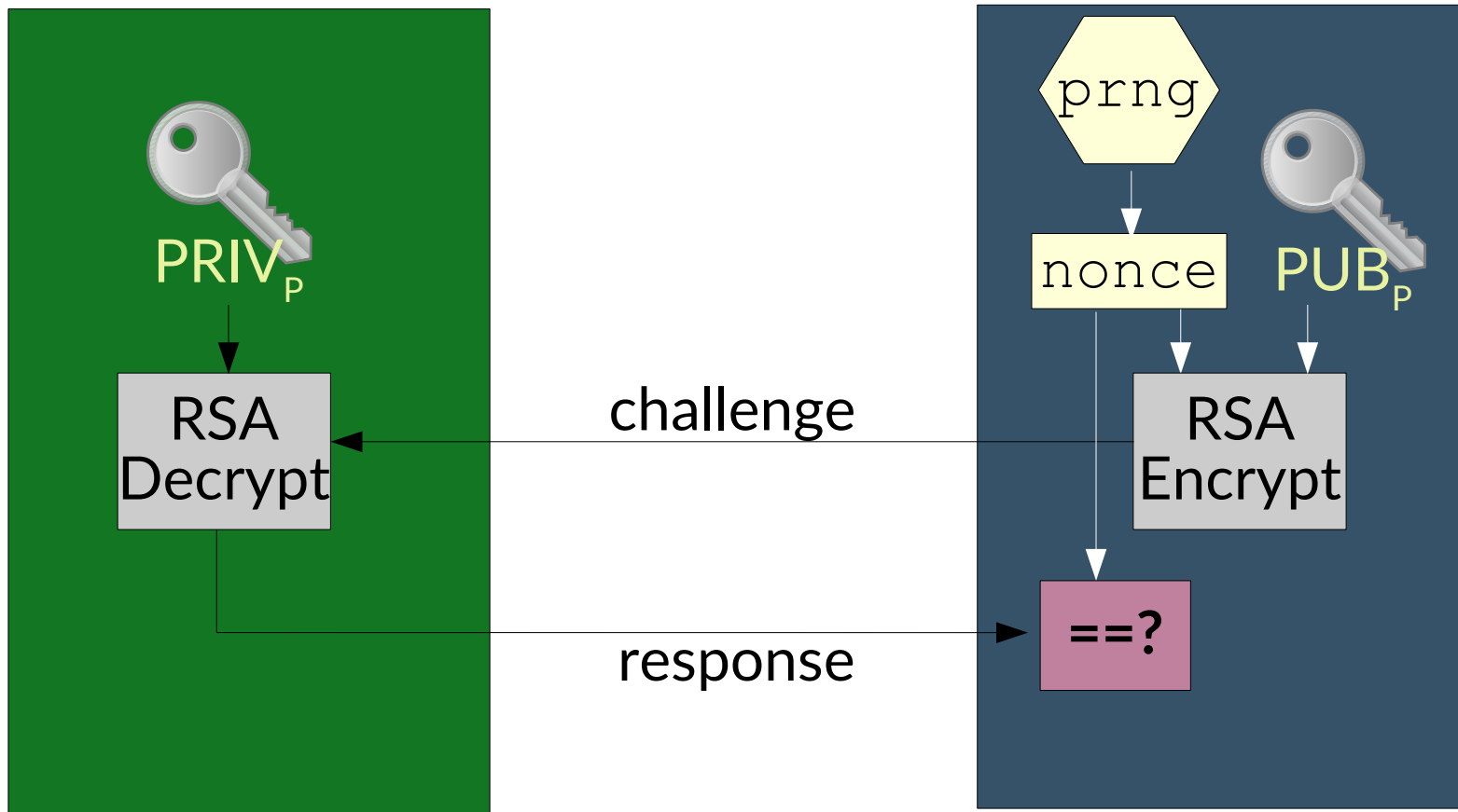
- P conosce il proprio segreto N
- V viene inizializzato col risultato dell'applicazione ripetuta k volte di una funzione hash a N: $h^k(N)$



- Alla prima autenticazione P invia $h^{k-1}(N)$
 - V verifica facilmente che $h(h^{k-1}(N)) = h^k(N)$
 - V scarta $h^k(N)$ e ricorda $h^{k-1}(N)$ come riferimento per la prossima autenticazione
- L'hash è
 - facile da calcolare → efficiente
 - difficile da invertire → sicuro
- Il sistema va però reinizializzato dopo k passi
 - ci sono varianti senza limiti

Sistemi a sfida e risposta

- Tipicamente utilizzati con crittografia asimmetrica
- **P** può provare il possesso di una chiave privata senza svelarla, se **V** possiede la chiave pubblica



2FA

- **Le credenziali vengono comunemente rubate tramite:**
 - Attacchi di phishing mirati.
 - Siti di terze parti compromessi con stesso nome utente / password utilizzati
 - Esempio Leak. Haveibeenpwned?
- **L'autenticazione a due fattori consiste nell'utilizzo di almeno DUE dei TRE (quattro?) fattori precedenti.**
- **L'autenticazione a due fattori aggiunge un ulteriore livello di autenticazione che impedisce agli aggressori di accedere a un account anche se ottengono le credenziali.**

Google Account



Password Checkup



We analyzed your saved passwords and found the following issues



46 compromised passwords



Change these passwords now

The following accounts use passwords which were exposed in a third-party data breach. Change these passwords immediately to keep your accounts safe. [Learn more](#)

2FA vs 2SA (Two steps authentication/verification)

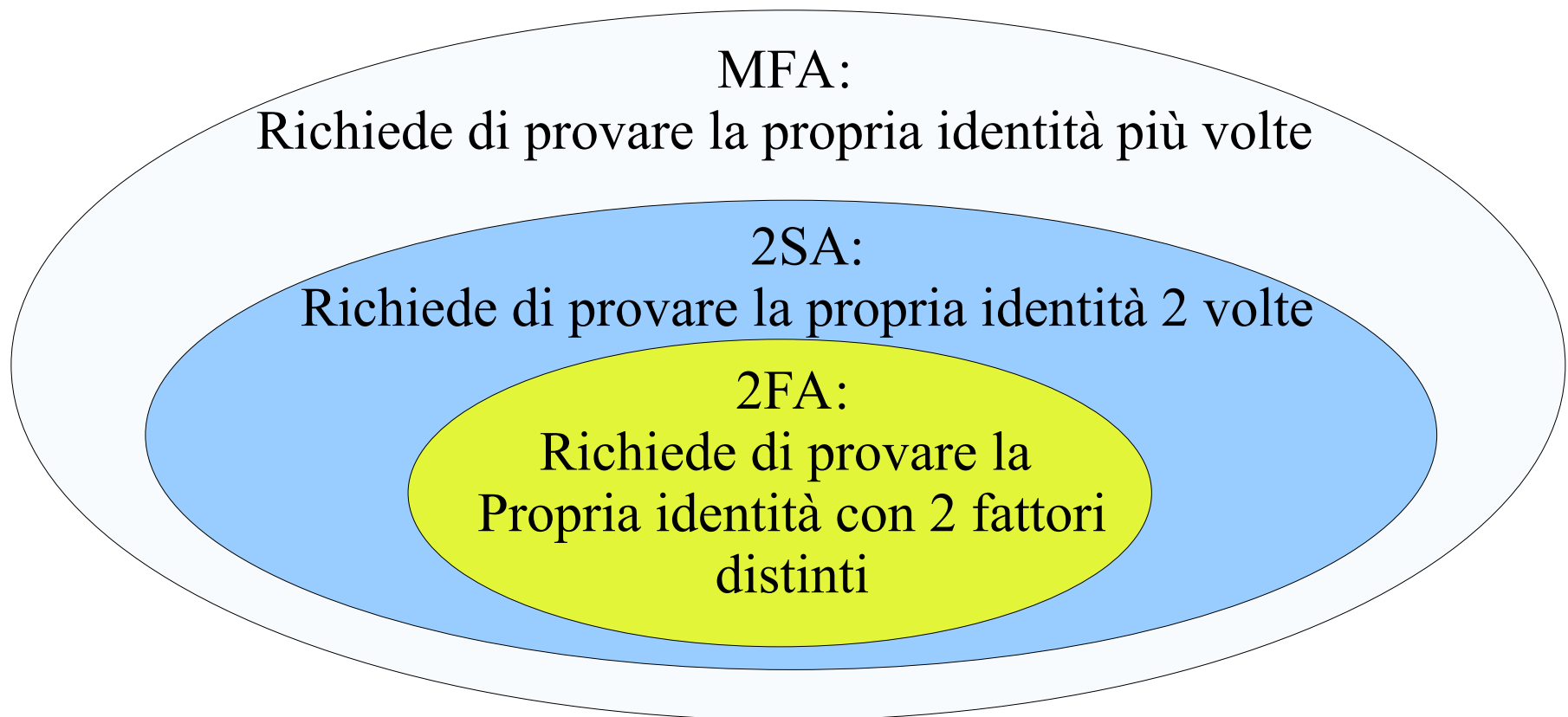
- È importante che per fare la 2FA i fattori di autenticazione siano **DISTINTI**
- Al giorno d'oggi infatti la 2FA viene erroneamente confusa con la two-steps, dove il livello aggiuntivo di autenticazione non è riconosciuto come distinto.
- Ad esempio un codice di verifica inviato per SMS o per mail in aggiunta ad una password non è considerato qualcosa che si **POSSIEDE** (in aggiunta a qualcosa che sa) perché l'SMS è la mail sono “facilmente” oggetto di attacchi MITM

2FA vs 2SA (Two steps authentication/verification)

- Implementare 2SA è comunque meglio di non implementarla, però è bene distinguerne le differenze con la 2FA
- Il device per l'autenticazione aggiuntiva deve essere dedicato solo a quello scopo. (Il telefono può essere violato da remoto invalidando il concetto del “possiede”)
- Sbloccare un'app con un PIN per avere un token di accesso aggiuntivo è considerato come “conoscenza aggiuntiva” per cui non un fattore di autenticazione aggiuntivo

2FA vs 2SA vs MFA

- È infine possibile considerare anche i casi dove più di due fattori, anche distinti, vengano usati per l'autenticazione, in quel caso parliamo Multi-factor Authentication



ESEMPI: OTP

- OTP (One Time Password) la password aggiuntiva, sottoforma di token è valida solo per un utilizzo.



Source: times.com

ESEMPI: TOTP

- TOTP (Time One Time Password) la password aggiuntiva, sottoforma di token è valida solo per un utilizzo ed è limitata nel tempo. Il tempo può variare dai 5 secondi a pochi minuti.



Source: ftsafes.com

STANDARDS

■ FIDO (Fast IDentity Online) Alliance

- Gruppo di aziende leader come Google e Microsoft che sviluppano standard per consentire un'esperienza di autenticazione più semplice e sicura su siti web e servizi mobile
- UAF (Universal Authentication Framework)
 - Progettato come sostituto dell'autenticazione di base
 - Tipicamente coinvolge la biometria in cui le informazioni di sicurezza non lasciano mai il dispositivo
- U2F (Universal Second Factor)
 - Rafforza e semplifica 2FA utilizzando dispositivi USB, NFC o Bluetooth
 - Ha come scope una forte protezione da phishing, dirottamento di sessioni, attacchi man-in-the-middle e malware
 - Supporto nativo offerto dai principali fornitori e browser

FIDO UAF

- **FIDO UAF supporta la possibilità di autenticazione senza password.**
 - È stato rilasciato come standard aperto dall'alleanza FIDO.
 - In questo standard, un utente che si autentica su un'applicazione o un servizio sfrutterà uno o più fattori di sicurezza sul proprio dispositivo digitale (solitamente un telefono cellulare) per rilasciare una chiave privata che viene utilizzata per firmare una sfida emessa dal server FIDO UAF.
 - Il meccanismo di verifica dell'utente sul dispositivo stesso può essere biometrico, basato sulla conoscenza o sul possesso per sbloccare la chiave privata per le funzioni di firma.

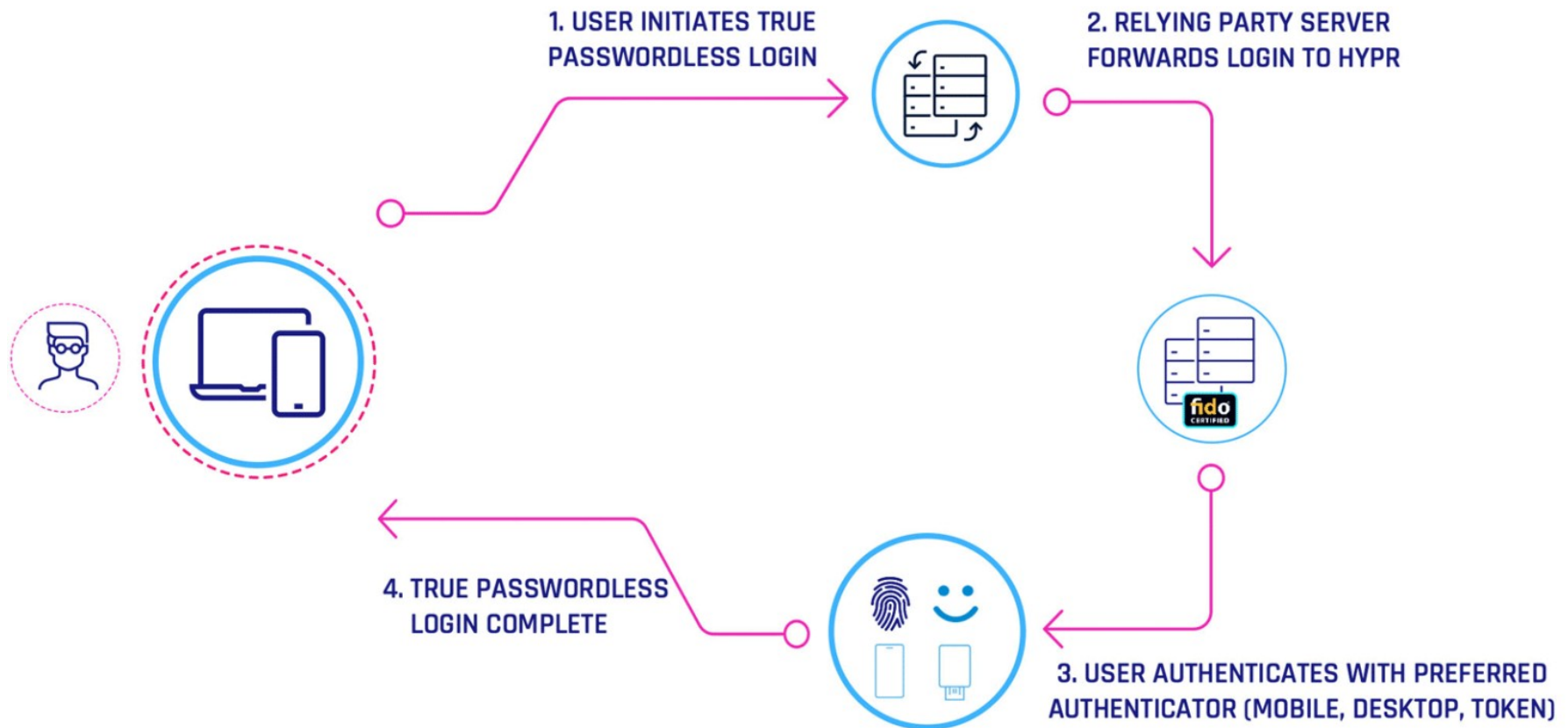
Fido Certified →
Source: hypr.com



FIDO UAF

■ FIDO UAF Architettura:

TRUE PASSWORDLESS WEB LOGIN



Source: hypr.com

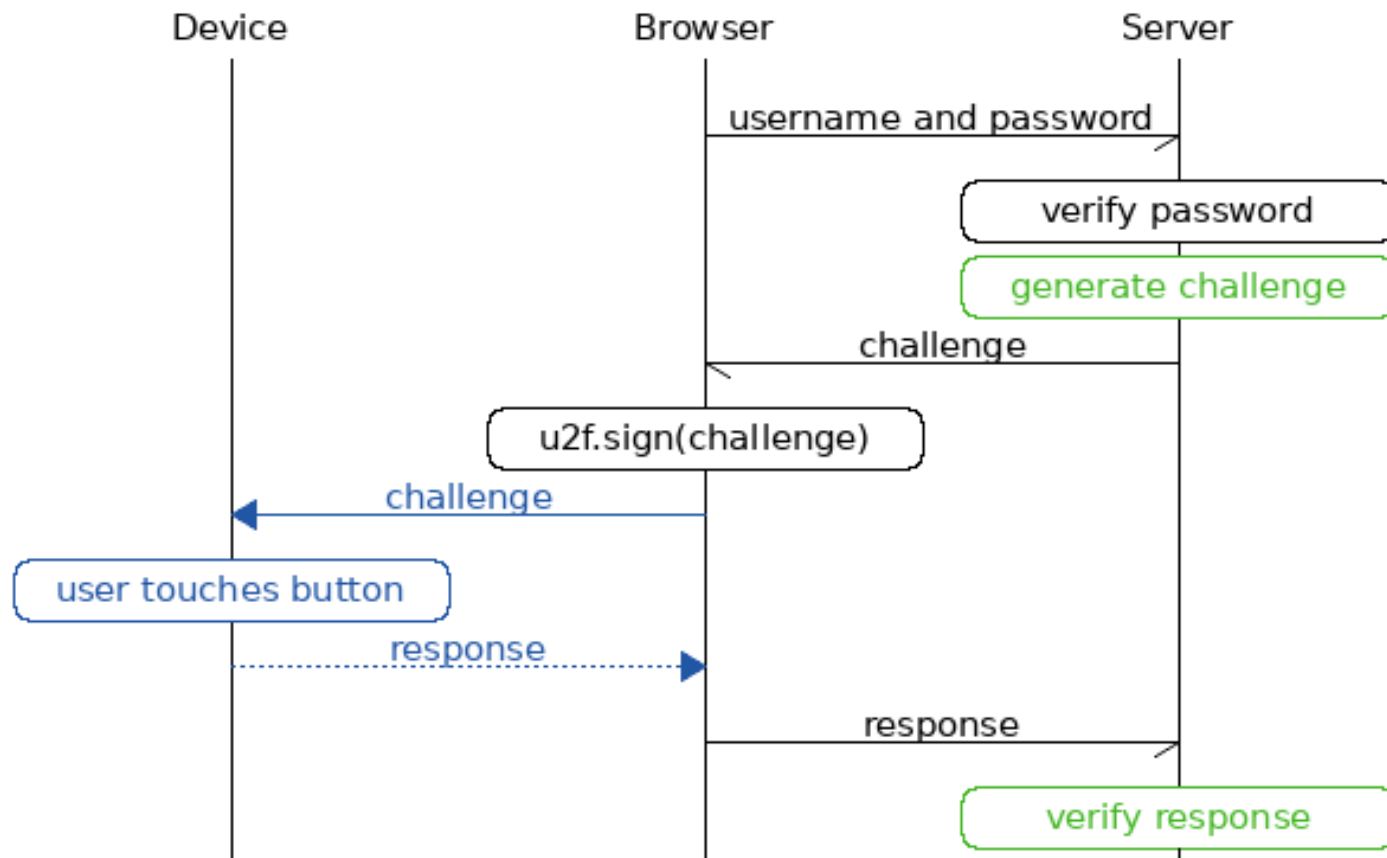
FIDO U2F

■ FIDO U2F.

- U2F è uno standard di autenticazione aperto che consente agli utenti di accedere in modo sicuro a qualsiasi servizio online con una singola chiave di sicurezza, istantaneamente e senza bisogno di driver o software client. FIDO2 è l'ultima generazione del protocollo U2F.
- U2F è stato creato da Google e Yubico, con il supporto di NXP, con l'intento di portare una forte crittografia a chiave pubblica nel mercato di massa. Oggi, le specifiche tecniche sono controllate dal consorzio del settore dell'autenticazione aperta FIDO Alliance.
- U2F è stato implementato con successo da servizi su larga scala, tra cui Facebook, Gmail, Dropbox, GitHub, Salesforce.com, il governo del Regno Unito e molti altri.

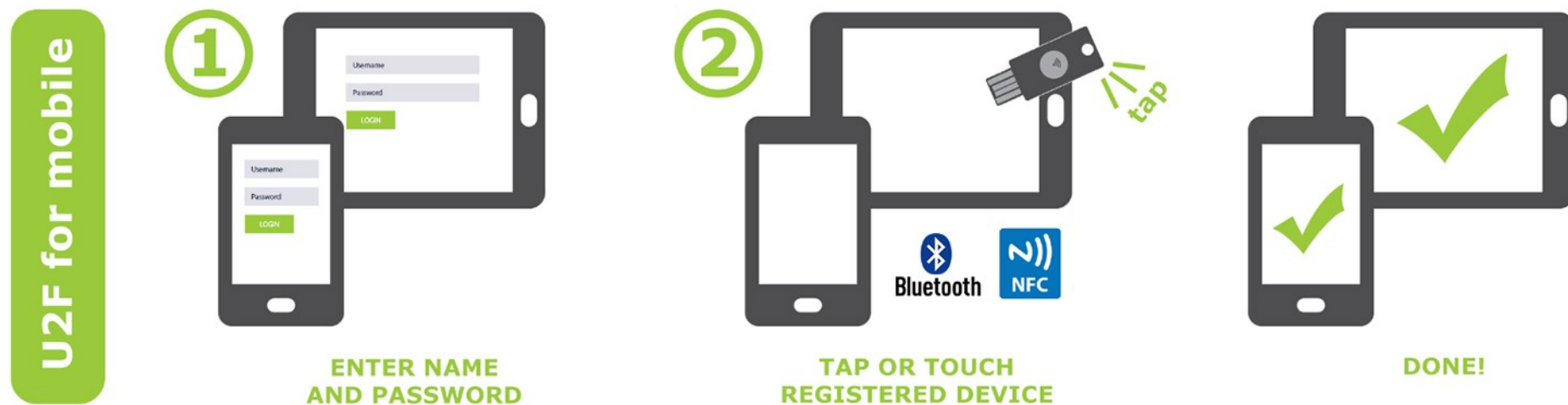
FIDO U2F

■ U2F Architettura implementativa. (U2F libreria ufficiale)



FIDO Casi d'uso

■ U2F per mobile

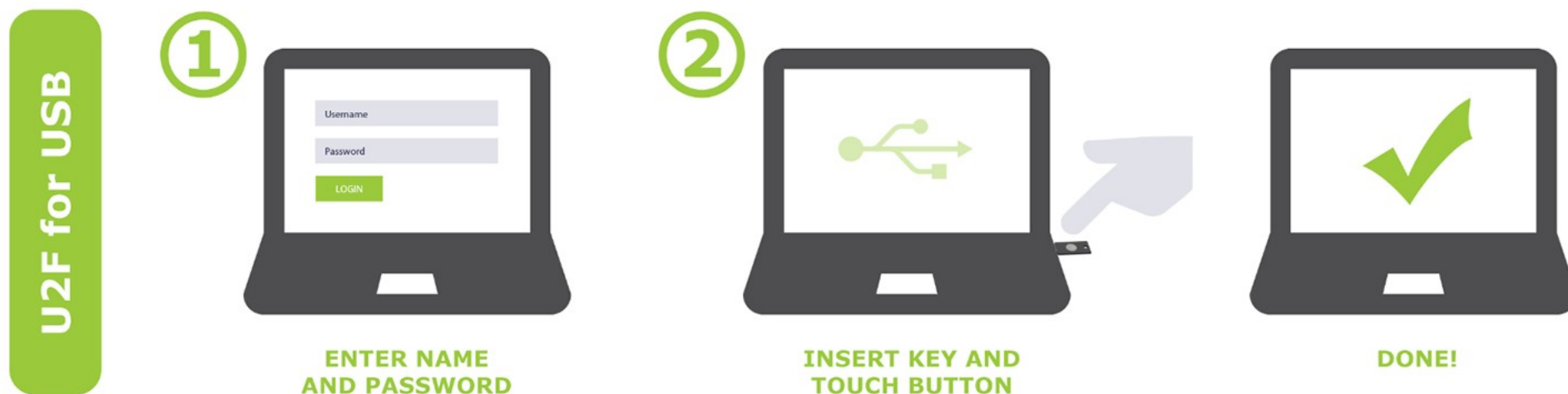


FIDO Casi d'uso

■ U2F per USB la YubiKey

- Componente hardware per 2FA attraverso OTP, e crittografia a chiave pubblica con supporto:

- Multi-protocol support; FIDO2/WebAuthn, U2F, Smart card, OpenPGP, OTP
- USB-A, USB-C, NFC



Siamo quindi al sicuro? MAI! 1/2



- In Chrome versione 61, Google ha introdotto una feature chiamata WebUSB. WebUSB permette ad un sito di fare delle richieste dirette a un device USB attraverso JavaScript.
- Considerando la potenza di JavaScript è teoricamente possibile che una pagina web possa interagire direttamente con un device USB se non correttamente isolato.
- Una funzionalità della YubiKey è quella dell' "origin-check" ed è usata per verificare la correttezza del sito nel quale si sta navigando, per evitare attacchi di phishing (verifica la legittimità del dominio)

Siamo quindi al sicuro? MAI! 2/2



- Il 16 febbraio 2018, i ricercatori Markus Vervier e Michele Orrù hanno dimostrato come aggirare il controllo dell'origine della YubiKey FIDO U2F utilizzando la feature WebUSB.
- Markus e Michele hanno mostrato come utilizzare WebUSB per passare le richieste U2F all'interfaccia CCID USB su YubiKey NEO, aggirando così il controllo dell'origine e creando un potenziale problema di sicurezza.
- Era quindi possibile “ingannare” la YubiKey sull'effettiva legittimità del sito andando a interagirci direttamente tramite WebUSB

Talk della presentazione a :

<https://www.youtube.com/watch?v=pUa6nWWTO4o>